Midterm 2013 – riešenia

Čas na písomku je 60 minút. Rozvrhnite si ho dobre. Najskôr sa sústreďte sa na hlavné myšlienky, potom prípadne doplňte detaily.

1 Formalizmus (10 bodov)

V tejto úlohe klaďte hlavný dôraz na presnosť a korektnosť práce s formalizmom z prednášky.

V tejto úlohe je povolené sa odvolať na definíciu množiny primitívne rekurzívnych funkcií a bez dôkazu použiť primitívnu rekurzívnosť funkcie $\mathrm{add}(x, y) = x + y$ a funkcie $\mathrm{sgn}(x) = [1 \text{ pre } x > 0,\ 0 \text{ pre } x = 0]$.

Vyriešte nasledujúce úlohy:

a) (2 body) Dokážte, že funkcia f daná predpisom $f(x) = 2x + 47$ je primitívne rekurzívna.

b) (4 body) Dokážte, že funkcia mul daná predpisom $\mathrm{mul}(x, y) = x \cdot y$ je primitívne rekurzívna.

c) (4 body) Formálne vyslovte a dokážte „vetu o if-e". (Slovne: Nech f je funkcia, ktorá vznikne tak, že podľa pravdivosti primitívne rekurzívneho predikátu počítame jednu alebo druhú primitívne rekurzívnu funkciu. Potom f je primitívne rekurzívna.)

2 Formalizmus (10 bodov) – riešenia

2.1 Podúloha A

Funkciu f vieme vyrobiť postupnou kompozíciou zo známych primitívne rekurzívnych funkcií:

$$\begin{aligned} & f_0 \equiv Comp[\mathrm{add}, P_1^1, P_1^1] \\ \forall i \in \{1, \dots, 47\}: \quad & f_i \equiv Comp[s, f_{i-1}] \end{aligned}$$

Potom f_{47} je hľadaná funkcia.

2.2 Podúloha B

Funkciu vieme vyrobiť primitívnou rekurziou. Vieme, že platí $\mathrm{mul}(0, y) = 0$ a $\mathrm{mul}(x+1, y) = \mathrm{mul}(x, y)+y$, potrebujeme teda dve pomocné funkcie: unárnu konštantnú nulovú funkciu z_1 a ternárnu funkciu $\mathrm{add}_{1,3}$, ktorá sčíta svoj prvý vstup (predchádzajúcu hodnotu funkcie mul) a tretí vstup (hodnotu y).

$$\begin{aligned} z_1 &\equiv PR[z, P_1^2] \\ \mathrm{add}_{1,3} &\equiv Comp[\mathrm{add}, P_1^3, P_3^3] \\ \mathrm{mul} &\equiv PR[z_1, \mathrm{add}_{1,3}] \end{aligned}$$

2.3 Podúloha C

Veta o if-e (základná podoba):

Nech f_1, f_2 a p sú primitívne rekurzívne funkcie rovnakej arity n. Potom je primitívne rekurzívna aj funkcia g definovaná nasledovne:

$$\forall x_1, \dots, x_n \in \mathbb{N}^n: \quad g(x_1, \dots, x_n) = \begin{cases} f_1(x_1, \dots, x_n) & \leftarrow p(x_1, \dots, x_n) > 0 \\ f_2(x_1, \dots, x_n) & \leftarrow \text{inak} \end{cases}$$

Dôkaz: Funkciu g vieme zapísať nasledovne: $g(\overline{x}) = \mathrm{sgn}(p(x)) \cdot f_1(x) + \overline{\mathrm{sgn}}(p(x)) \cdot f_2(x)$. Formálne teda

$$\begin{aligned} g_1 &\equiv Comp[\ \mathrm{mul},\ Comp[\mathrm{sgn}, p],\ f_1\ ] \\ g_2 &\equiv Comp[\ \mathrm{mul},\ Comp[\overline{\mathrm{sgn}}, p],\ f_2\ ] \\ g &\equiv Comp[\ \mathrm{add}, g_1, g_2\ ] \end{aligned}$$

No a pre poriadok ešte (sorry, zabudol som dať $\overline{sgn}$ do zadania):

$$\begin{aligned} j &\equiv Comp[s, z] \\ z_2 &\equiv Comp[z_1, P_1^2] \\ \overline{sgn} &\equiv PR[j, z_2] \end{aligned}$$

3 Známa pôda (5+5 bodov)

V tejto úlohe sa pýtam na odprednášané veci. Riešte ich bez odvolávania sa na prednášky. V podúlohe a) stačí konštrukciu popísať dostatočne podrobne slovne, podúlohu b) robte pokiaľ možno formálne.

a) Dokážte, že trieda jazykov, ku ktorým existuje Wangov dlaždicový program používajúci len jeden riadok dláždenia, je presne rovná triede regulárnych jazykov neobsahujúcich prázdne slovo.

b) Nech φ je binárna totálna funkcia s veľkými funkčnými hodnotami. Dokonca až tak veľkými, že ku každej unárnej rekurzívnej funkcii f existuje konštanta c_f taká, že $\forall n \geq c_f : \varphi(c_f, n) > f(n)$. Môže byť φ rekurzívna?

4 Známa pôda (5+5 bodov) – riešenia

4.1 Podúloha A

Ku každému regulárnemu jazyku R existuje DKA A, ktorý ho rozpoznáva. Ak $\varepsilon \notin R$, tak vieme A prerobiť na NKA A' akceptujúci R pomocou jedného akceptačného stavu. Na to stačí pridať nový (a odteraz jediný) akceptačný stav f a navyše vždy, kedy pôvodný automat išiel do pôvodného akceptačného stavu, bude mať nový automat nedeterministicky na výber dve možnosti – buď spraví to isté ako pôvodný DKA, alebo prejde do f.

No a A' vieme priamo prerobiť na príslušnú sadu dlaždíc. Ľavá stena dostane ako farbu začiatočný stav, pravá stena f, spodná stena $\heartsuit$, no a každému riadku δ-funkcie automatu A' bude prislúchať jeden typ dlaždíc: ak $q \in \delta_{A'}(p, x)$, tak budeme mať dlaždicu (vľavo p, hore x, vpravo q, dole $\heartsuit$).

No a naopak. Ku každému dlaždicovému programu ľahko zostrojíme NKA, ktorý postupne zľava doprava nedeterministicky háda priložené dlaždice a kontroluje, či na seba pasujú. (Konštrukcia je v podstate presným obrátením predchádzajúcej, preto ju neuvádzame.)

4.2 Podúloha B

Sporom. Nech je φ rekurzívna.
Potom je rekurzívna aj $\varphi_1 \equiv Comp[\varphi, P_1^1, P_1^1]$, teda unárna funkcia, pre ktorú platí $\forall x : \varphi_1(x) = \varphi(x, x)$.
Ale ak je rekurzívna φ_1, tak k nej existuje jej konštanta c_{φ_1} (ďalej len c) taká, že $\forall n \geq c : \varphi(c, n) > \varphi_1(n)$.
No a to už je očividný spor, lebo pre $n = c$ dostávame $\varphi(c, c) > \varphi_1(c) = \varphi(c, c)$.

5 Exkurzia do neznáma (10 bodov)

V tejto úlohe sa pýtam na niečo, čo som neprednášal. Pri riešení sa smiete odvolať na čo len chcete. Prípadné konštrukcie stačí popisovať slovne, netreba formálne.

Uvažujme nasledovný model vypočítateľnosti: Deterministický Turingov stroj s jedinou, obojsmerne nekonečnou páskou. Na rozdiel od klasického TS je táto páska **read-only** (celá, ani blanky sa nedajú prepisovať). Zato však máme k dispozícii až **tri** hlavy. Na začiatku výpočtu sú všetky hlavy na prvom písmene vstupného slova. (Rovnako ako pri klasických viachlavých TS pri každom kroku výpočtu stroj prečíta symboly pod všetkými hlavami a podľa nich a aktuálneho stavu zmení stav a pohne každou z hláv o najviac jedno políčko.)

Rozhodnite a dokážte, či je tento model vypočítateľnosti Turingovsky úplný, teda presne rovnako silný ako klasické TS.

Hint: Ako vyzerá konfigurácia takéhoto TS? Poznáte iné Turingovsky úplné modely vypočítateľnosti s podobne vyzerajúcou konfiguráciou? Ako súvisia s týmto?

Záchrana za 3 body: ak neviete úlohu vôbec riešiť, zvoľte si nejaký neregulárny jazyk a zostrojte k nemu takýto TS, ktorý ho bude rozpoznávať.

6 Exkurzia do neznáma (10 bodov) – riešenie

Takýto stroj vieme simulovať na klasickom TS, preto určite nie je silnejší. (Toto netreba zabudnúť explicitne uviesť.)

Na opačnú inklúziu si stačí uvedomiť, že ľubovoľnú konkrétnu hlavu (presnejšie, jej polohu) vieme použiť ako počítadlo. Jeho hodnota bude vzdialenosť hlavy smerom doľava od prvého písmena vstupu. Inkrementovanie počítadla je pohyb hlavy doľava, test na 0 je pozrietie, či vidím blank alebo písmeno, a dekrementovanie je pohyb doprava. Dve z troch hláv vieme teda použiť ako počítadlá a navyše máme ešte pre pohodlie tretiu na začiatku na prečítanie vstupu. Z prednášok vieme, že takýto model je Turingovsky úplný. (Pri vhodnej definícii vstupu a výstupu by dokonca stačili aj dve hlavy, ale s tromi je to očividnejšie.)

7 Bonus (5 bodov)

Ak by sme TS z predchádzajúcej úlohy namiesto troch hláv nechali len jednu (a stále mal len read-only pásku), ako presne by sa zmenila trieda rozpoznateľných jazykov?

8 Bonus (5 bodov) – riešenie

Takéto TS rozpoznávajú len regulárne jazyky. V podstate ide o dvojsmerné konečné automaty, len navyše vedia chodiť na výlety po blankoch naľavo/napravo od vstupu. Ku každému takémuto TS vieme ale zostrojiť ekvivalentný 2DKA: z prechodovej funkcie vyhádžeme pravidlá na chodenie po blankoch a namiesto nich pridáme nové, ktoré na jeden krok odsimulujú celú prechádzku po blankoch. Ak teda v pôvodnom DTS bolo napr. pravidlo „keď si na blanku v stave p, posuň sa tam-a-tam a zmeň stav na taký-a-taký“, bolo jednoznačne určené, čo sa bude diať, ak sa toto pravidlo použije na prvom blanku napravo od slova. Boli len dve možnosti: buď sa niekedy hlava vrátila spät na slovo, a to v nejakom stave q, alebo sa tak nikdy nestalo. V prvom prípade bude náš 2DKA mať nové pravidlo „keď si na zarážke za slovom v stave p, posuň sa doľava a zmeň stav na q“, v druhom prípade sa náš 2DKA v dotyčnej situácii zasekne a vo výpočte nepokračuje.